Tools for educational data mining

Tools for educational data mining: a review

Slater, S., Joksimovic, S., Kovanovic, V., Baker, R.S., Gasevic, D.

Abstract. In recent years, a wide array of tools have emerged for the purposes of conducting educational data mining (EDM) and/or learning analytics (LA) research. In this article, we hope to highlight some of the most widely used, most accessible, and most powerful tools available for the researcher interested in conducting EDM/LA research. We will highlight the utility that these tools have with respect to common data preprocessing and analysis steps in a typical research project as well as more descriptive information such as price point and user-friendliness. We will also highlight niche tools in the field, such as those used for Bayesian knowledge tracing (BKT), data visualization, text analysis, and social network analysis. Finally, we will discuss the importance of familiarizing oneself with multiple tools—a data analysis toolbox—for the practice of EDM/LA research.

1 Introduction

In recent years the educational data mining (EDM) and learning analytics (LA) communities have emerged as alternatives to frequentist and Bayesian approaches for working with educational data (Romero & Ventura, 2007; Baker & Siemens, 2014). Data mining, also referred to as knowledge discovery in databases (KDD), involves methods that search for new and generalizable relationships and findings, rather than attempting to test prior hypotheses (cf. Collins et al., 2004). While some statisticians use the term “data mining” in a pejorative fashion, to indicate the unprincipled search for hypotheses and pretense that these hypotheses were investigated in isolation, data mining as an area of methods has an extended history going back to exploratory data analysis (Tukey, 1977) and has established methods for determining validity and generalizability. The EDM and LA communities build off the long-standing traditions of data mining and analytics in other fields, such as bioinformatics (Wang et al., 2005) and data mining for business (Berry & Linoff, 1997).

This paper will discuss some of the tools that have emerged for research and practice in educational data mining, discussing where relevant tools also used by the broader data mining and data science communities. This paper will not provide a general review of methods in educational data mining -- for that, see reviews in (Romero & Ventura, 2010; Baker & Siemens, 2014); and see extended discussions of methods in (Baker, 2015).

This paper’s review will focus on educational data mining tools, and tools frequently used to conduct educational data mining analyses, rather than the broader universe of tools used to conduct more traditional and modern statistical analyses. For example, tools for creating structural equation models and multilevel models will not be covered. Other reviews of these

types of tools have been published, often in this very journal (cf. Preacher, Curran, & Bauer, 2006). Similarly, general data management tools such as database management systems will also not be covered, except where they offer particularly relevant functionality. The inclusion criteria for this article will be somewhat informal; rather than attempting to cover every tool ever created that *could be* used for EDM, or every tool ever created and used by a single research group, we will cover the primary tools used by some of the core research groups and/or organizations in the field. This necessarily means that a specific researcher's favorite tool may be excluded; we nonetheless hope that this review will provide useful information to researchers new to this area of methods on what tools they may find useful.

2 Overview of the important EDM/LA tools

In this paper we will attempt to detail the most widely used, most accessible, and most powerful tools available to the EDM or LA researcher or practitioner. The course of this discussion will follow, roughly, the path one might take while exploring a research question or conducting an analysis. In educational data mining, as well as in other areas of data mining and data science, transforming raw and inchoate data streams into meaningful variables is the first major challenge in the process. Often data come in forms and formats that are not ready for analysis; the data not only need to be transformed into a more meaningful format but in addition meaningful variables need to be engineered (see section 3.3 in Baker, 2015, or Veeramachaneni et al., 2015, for a more thorough discussion of this process). In addition, data often need to be cleaned to remove cases and values that are not simply outliers but actively incorrect (i.e. cases where time-stamps have impossible values, instructor test accounts in learning system data, etc.). We will begin with an overview of two tools well-suited for the manipulation, cleaning, and formatting of data, as well as for feature engineering and data creation: Microsoft Excel, Google Sheets, and the EDM Workbench. We will also discuss the programming language Python, and database queries, for the role that they play in a programmatic approach to this particular task.

After data cleaning, transformation into a more workable format, and feature engineering, the next question facing an EDM or LA researcher is that of analysis - what tests can be conducted, what models can be constructed, what relationships can be mapped and explored, and how can we validate our findings? We discuss a set of tools that are appropriate for this task: RapidMiner, Weka, KEEL, KNIME, Orange, and SPSS. We also identify several packages in Python which are well-suited for testing, analysis, and modeling.

The tools mentioned so far are relevant to a range of types of data and analysis. However, some types of data can be more effectively analyzed with more specialized tools tailored to those domains. We will discuss tools frequently used in educational data mining for these types of specialized data, including implementations of knowledge tracing algorithms, text mining, social network analysis, sequence mining, and process mining. We do not present these specific cases in an attempt to be exhaustive, rather, we discuss them because of their current popularity to researchers and practitioners.

Once a researcher has conducted analyses and has a validated, well-performing model, that work is often then shared with other researchers, analysts, and practitioners in schools and universities or in curriculum development. A crucial component of the distribution of research is legible and informative visualizations, and in the last portion of our discussion we will cover a selection of tools that afford data scientists the ability to create polished and informative graphs, charts, models, networks, diagrams, and other manners of visualized information. We identify three visualization tools: Tableau, d3js, and InfoVis, as well as discuss the potential for visualization through a handful of popular Python packages.

As our final featured tool, we discuss the PSLC DataShop, which is a unique tool that integrates data collection, construction, analysis, and visualization. DataShop affords the researcher an ability to conduct a set of analyses popular among cognitive scientists and EDM researchers with one tool.

2.1 Data manipulation and feature engineering

Before data mining can be conducted, datasets must first be cleaned and prepared from their raw state. While this problem is usually present with any data, data miners typically work with messier data than statisticians and psychometricians; instead of meaningfully recorded test or survey data, data miners often work with log data or learning management system data recorded in forms that are not immediately amenable to analysis. Readers with experience working with these types of educational data know that it is messy, sometimes incomplete, sometimes in several parts that must be merged, and occasionally in unfamiliar or inconvenient formats. A researcher may be interested in analyzing students, but their data may consist of system-logged actions. A researcher may be interested in utilizing durations between actions to identify off-task students (e.g. Baker, 2007; Cetintas et al., 2010), but only have access to raw timestamps. In these situations, new variables must be created in order to conduct the desired analyses, a process termed *feature engineering* (Baker, 2015; Veeramachaneni et al., 2015).

We present the following tools that can be used for cleaning, organizing, and creating data. We will discuss the merits of each tool, discussing their utility for manipulating and restructuring large datasets; and for creating/engineering new and more useful variables from existing variables.

Microsoft Excel/Google Sheets.
Microsoft Excel is easily the most accessible tool for data scientists interesting in manipulating or engineering data, and does a great job of making the data easily visible as it is edited. It has been joined recently by Google Sheets, a similar web-based tool. These tools are not useful for engineering variables in extremely large data sets, around one million rows and above, but they are excellent tools for smaller-scale feature engineering, and for prototyping new variables in subsets of a much larger data set.

One of the key reasons for their usefulness for first-stage analysis and prototyping of new data features (variables) is that Excel and Sheets are good at presenting data clearly within a fully

visual interface. This makes it easy to identify structural or semantic problems in the data, such as unusual or missing values, or duplicate entries. These tools also make it very straightforward to engineer new features, rapidly apply these features to the entire sheet, and visually check the features across a range of data for appropriate functioning. Summaries of students, problems, and problem sets, as well as other aggregations, can be easily calculated through filters and summations or through pivot tables, and there is functionality for linking between data sets or levels of aggregation.

At the same time, Excel and Sheets are not ideal for all types of feature creation. Creating features requiring different aggregations of the data can involve sorting and re-sorting the data several times, making it challenging to keep records of what was done, and making it easy to accidentally change feature semantics. More importantly, Excel and Sheets have limits on the amount of data that can be loaded and manipulated and still maintain reasonable performance. Several common operators in Excel and Sheets can reduce performance further.

EDM Workbench.

The EDM Workbench (Rodrigo et al., 2012), available for free download at http://penoy.admu.edu.ph/~alls/downloads-2, is a tool for automated feature distillation and data labeling. Much of the automated feature distillation functionality of EDM Workbench is addressed at specific shortcomings of Excel and Sheets for specific tasks of relevance to data scientists, such as the generation of complex sequential features, data sampling, labeling, and the aggregation of data into subsets of student-tutor transactions based on user-defined criteria (referred to as 'clips'). The EDM Workbench enables researchers to create features through xml-based authoring, and also has built-in functionality to distill a set of 26 features used in existing literature and intelligent tutoring systems. The features include (but are not limited to) the time the student spent on the problem (both in absolute and relative terms – for instance how much faster or slower the student was than other students working on the same problem step); and the types, number and proportion of correct, wrong, or help actions for the current skill for the last n steps, for the skill, or for the student.

In terms of data labeling, the EDM Workbench has functionality for creating text replays (Baker, Corbett, & Wagner, 2006), pretty-printed segments of human behavior that are coded by researchers or other domain experts in terms of categories of behavior or other labels of interest. The EDM Workbench supports sampling, inter-rater reliability checking, and synchronization between labels and features distilled.

Python and Jupyter notebook

For data scientists with programming knowledge, there are a handful of languages that are particularly suited to the manipulation of data and engineering of features. Python is considered by many to be a particularly useful language for these purposes. In particular, engineering context-dependent or temporal features is easier in Python than in Excel or Google Sheets. Another useful feature of Python is Jupyter notebook – a server-client application that allows for the creation and modification of Python code and rich text elements such as graphs and tables within a web browser. Jupyter notebook is a method for keeping a record of analyses

conducted and intermediate results, displaying each user action and its result, in order. However, despite this advantage, it is still easier to visually inspect data and features created in Excel or Google Sheets. Missing data, duplicate cases, or unusual values can be especially difficult to identify in datasets, and validation of engineered features can be more time-consuming, especially for novice programmers. Additionally, Python is able to handle many different types of unusual or specialized data formats, such as the JavaScript Object Notation (JSON) files produced by several MOOC and online learning platforms. While Python is computationally more powerful than the spreadsheet tools covered earlier, its capacities in these areas is not infinite. While Python is able to accommodate larger datasets than previous tools, it is still subject to size limitations, becoming slower at around the range of 10 million rows of data for these researchers' computers. It is important to note that some types of programs (for example, those involving nested loops) are significantly slower when using the notebook than in standard Python.

SQL

SQL, or the Structured Query Language, is used to organize some databases. SQL queries can be a powerful method for extracting exactly the desired data, sometimes integrating ("joining") across multiple database tables. Many basic filtering tasks, such as selecting a specific subset of students, or obtaining data from a specific date range, are significantly faster in database languages such as SQL than in any of the tools mentioned above. However, SQL can be a somewhat clunky language for the creation of complex features in the feature engineering process. SQL can work effectively in combination with the other aforementioned tools: SQL excels at the bulk sorting and filtering tasks that are very slow in Excel or Python, while these tools perform better on the kinds of reduced size datasets that SQL is able to produce.

2.2 Algorithmic analysis

Once features have been engineered, outcome variables and ground truth have been labeled, and data has been sampled and structured appropriately for analysis, the next step is to begin analysis and modeling of the dataset, and validate the resulting models. The tools listed in the following section offer a wide range of algorithms and modeling frameworks that can be used to model and predict processes and relationships in educational data.

RapidMiner

RapidMiner (http://rapid-i.com/content/view/181/190/) is a package for conducting data mining analyses and creating models. It has limited functionality for engineering new features out of existing features (such as the creation of multiplicative interactions), and for feature selection (based on inter-correlation of features with one another and with outcome measures). However, RapidMiner has an extremely extensive set of classification and regression algorithms, as well as algorithms for clustering, association rule mining, and other applications. Other algorithms can often be composed out of the operators contained in RapidMiner – for instance, to conduct ensemble selection or model bagging. Support for resampling methods such as bootstrapping, however, is more limited than in other data mining packages.

RapidMiner's graphical programming language is relatively more powerful than those of most other data mining tools, with considerable functionality for user specification. For instance, RapidMiner can be used to conduct cross-validation at multiple levels using the BatchCrossValidation operator. This support can be extremely useful for generalizability analyses and is an advantage over the graphical languages in most other data mining packages. RapidMiner also has a wide range of metrics available for model assessments, and can display visualizations such as Receiver-Operating Curves to help a user evaluate model fit. Models can be output either in terms of the actual mathematical models or in xml files which can be used to run the model on new data using RapidMiner code. A range of tasks that cannot be achieved in RapidMiner's graphical programming language can be achieved through its Application Program Interface (API) which can be integrated into programs written in Java or Python. RapidMiner incorporates all of the algorithms available in Weka, discussed below. Newer versions of RapidMiner also include crowd-sourced algorithm and parameter suggestions.

RapidMiner has an extensive set of tutorials which are very useful in learning how to use the graphical programming language. RapidMiner is available for free for academic use, and commercial licenses are available through the publisher Rapid-I.

WEKA

The Waikato Environment for Knowledge Analysis (Weka, http://www.cs.waikato.ac.nz/ml/Weka/) is a free and open source software package that assembles a wide range of data mining and model building algorithms. It does not support the creation of new features, though it does have support for automatic feature selection.

Weka has an extensive set of classification, clustering, and association mining algorithms that can be used in isolation or in combination, through methods such as bagging, boosting, and stacking. Users can invoke the data mining algorithms from the command line, a GUI (graphical user interface), or through a Java API. The command line interface and APIs are more powerful than the GUI, which does not give users access to all advanced functions. Weka can output the models it generates either in terms of the actual mathematical models, or in PMML (Predictive Modeling Markup Language) files which can be used to run the model on new data using the Weka scoring plugin to run the model.

Learning to use Weka is supported by a book by Witten, Frank & Hall (2011), now in its third edition. The Weka website also hosts an active mailing list, wiki, and bug reports.

SPSS

Like Excel, SPSS is known beyond just the data science community. SPSS is primarily a statistical package, and offers a range of statistical tests, regression frameworks, correlations, and factor analyses. SPSS is complemented by IBM SPSS Modeler Premium, a relatively newer analytics and data mining package which integrates previous analytics and text mining packages.

SPSS Modeler specifically has functionality for creating new features out of existing features, for data filtering, and for feature selection and feature space reduction. The tools for data transformation, feature selection, and feature space reduction are comparable to those seen in data mining packages, with a lower variety of selection approaches. There is also functionality for using the target class in feature selection, which is not available in many other packages.

While SPSS represents a comprehensive statistical analysis tool, support for modeling is somewhat worse than the other tools in this section. SPSS is less flexible than other tools, more difficult to customize, and is not documented as well. Support for procedures considered key by researchers in the educational data mining community, such as cross-validation, is also lacking when compared to tools more focused on data mining. SPSS is available commercially at http://www.ibm.com/analytics/us/en/technology/spss/.

KNIME

KNIME ("naim", KoNstanz Information MinEr, www.knime.org), formerly Hades, is a data cleaning and analysis package generally similar to RapidMiner and Weka. It offers many of the same capabilities as those tools, and like RapidMiner, incorporates all of Weka's algorithms. Additionally, it offers a host of specialized algorithms in areas such as sentiment analysis and social network analysis. An especially powerful aspect of KNIME is its ability to integrate data from multiple sources (e.g. a .csv of engineered features, a word document of text responses, and a database of student demographics) within the same analysis. KNIME also offers extensions that allow it to interface with R, Python, Java, and SQL.

Orange

Orange (orange.biolab.si) is a data visualization and analysis package. While it has considerably fewer algorithms and tools than RapidMiner, Weka, or KNIME it has a cleaner and easier to understand interface, with color-coded widgets differentiating between data input and cleaning, visualization, regression and clustering. It offers many commonly-used algorithms, such as k-nearest neighbors, random forests, naïve Bayes classification, and support vector machines. Orange also has customizable visualization modules for the presentation of model results with reasonable documentation. However, Orange is somewhat limited in the scale of data that it can process, comparable to Excel. Based on its easily understood GUI and menu layout, Orange may be better suited as a tool for smaller projects or more novice researchers.

KEEL

KEEL (http://sci2s.ugr.es/keel/) is a data mining tool used by many EDM researchers. Unlike some of the tools listed above, which attempt to broadly survey different types of methods, KEEL has extensive support for some types of algorithms and tasks, but limited support for other algorithms and tasks. For instance, KEEL has extremely extensive support for discretization algorithms, but has limited support for other methods for engineering new features out of existing features. It has excellent support for feature selection, with a wider range of algorithms than any other package. It also has extensive support for imputation of missing data, and considerable support for data re-sampling.

For modeling, KEEL has an extensive set of classification and regression algorithms, with a large focus on evolutionary algorithms (although it is worth noting that evolutionary algorithms are currently not favored by many/most EDM researchers). Its support for other types of data mining algorithms, such as clustering and factor analysis, is more limited than other packages. Support for association rule mining is decent, though not as extensive as some other packages.

KEEL has relatively less support for new users than most other data mining packages, though there are help features and a user manual. KEEL is open-source and free for use under a GNU license.

Spark MLLib
Spark (http://spark.apache.org/mllib/) is a framework for large-scale processing of data across multiple computer processors, in a distributed fashion. Spark can connect with several programming languages, including Java, Python, and SQL, through an API, allowing these languages to be used for distributed processing. Spark's MLLib machine learning framework provides implementations of several standard machine learning and data mining algorithms. Though MLLib's functionality is still somewhat limited, and it is a purely programmatic tool, its distributed nature makes it an efficient and rapid choice.

2.3 Visualizations

Beyond simply mining data, there is an increasing awareness that good visualization methods can support both analysts and practitioners in deriving meaning from data (Siemens and Baker, 2014, Duval, 2011, Verbert et al., 2013, Tervakari et al., 2014). In the next section, we discuss specialized tools for applications such as social network analysis that can provide sophisticated visualizations (e.g. Gephi, SNAPP). Specifically, we aim to introduce some of the general tools and methods for visual analytics, which enable building interactive visual interfaces for gaining knowledge and insight from data, as well as communicating important implications for learning to students and teachers.

Tableau
Tableau presents a family of products for interactive data analysis and visualization. Although the primary focus of the Tableau toolset is support for business intelligence, it has been commonly applied in educational settings to analyze student data, provide actionable insights, enhance teaching practices and streamline educational reporting.

The main advantage of Tableau is that no programming knowledge is needed to analyze large amounts of data from various sources, making a range of visualizations easily available to a wider community. Tableau provides functionality to connect or import data from several standardized formats for data storing (e.g., databases, data warehouses, log data). Tableau also has functionality for building rich and interactive dashboards, capable of displaying dynamic real-time visualizations to end users. However, Tableau's functionality is limited to this; it does not support predictive analytics or relational data mining. Moreover, Tableau, as a commercial tool, is not extendable and does not support integration with other software platforms. Tableau is available at www.tableau.com.

D3js

D3.js (Data Driven Documents; www.d3js.org) is a JavaScript library that allows manipulation of data-driven documents, enabling researchers and practitioners to build complex, interactive data visualizations that require data handling and are targeted for modern web browsers.

D3.js has several benefits; it allows considerable flexibility in building a range of kinds of data visualization, does not require installation, supports code reuse, and is free and open source. However, there are challenges to wider adoption for educational research purposes. As a technology, D3.js requires extensive programming knowledge and has compatibility issues as well as some performance limitations for larger data sets. Finally, it does not provide any means to hide data from users of visualizations, requiring data pre-processing to ensure privacy and data security.

Beyond D3.js, many other programmatic data visualization tools exist, aimed at providing different ways to present data visually and build interactive dashboards. Some of the commonly used tools include Chart.js, Raw, JavaScript InfoVis Toolkit, jpGraph, and Google Visualization API (see www.creativeblog.com/design-tools/data-visualization-712402 for further discussion). These tools offer broadly similar functionality to D3.js but have been less frequently used by EDM and LA researchers.

2.4 Specialized EDM and LA Applications

In the previous section, we discussed general-purpose tools for EDM modeling and analysis. However, specific types of data and specific analysis goals often require more specialized algorithms that are not available in these general-purpose tools. For these cases, researchers and practitioners typically use more specialized tools designed for these situations. In our last group of surveyed tools, we will cover the functionality of some of the most popular tools that accomplish these goals.

2.4.1 Tools for Bayesian Knowledge Tracing

Bayesian Knowledge Tracing (Corbett & Anderson, 1995), or BKT, is a popular method for latent knowledge estimation, where a student's knowledge is measured during online learning. This is distinct from the type of educational measurement common within tests in that, during online learning, the knowledge is changing while it is being measured.

Bayesian Knowledge Tracing is a Hidden Markov Model and simultaneously, a simple Bayesian Network (Reye, 2004) that predicts whether a student has or has not mastered a particular skill within an intelligent tutoring system or similar program. BKT models are typically fit using one of two algorithms: brute force grid search, or expectation maximization (EM). The two algorithms perform comparably in terms of predictive performance. Some of the publicly available tools for

BKT include BKT-BF, available at http://www.columbia.edu/~rsb2162/BKT-BruteForce.zip , BNT-SM, available at http://www.cs.cmu.edu/~listen/BNT-SM/ (also requires Matlab to run), and hmmsclbl, available at http://yudelson.info/hmmsclbl.html .

2.4.2 Text Mining

Text mining is a rapidly growing area of data mining and there are a significant number of programs, apps, and APIs available for the tagging, processing, and identification of textual data. Text analysis tools can process text parts of speech, sentence structure, and semantic word meaning. Additionally, some tools are able to identify representational relationships between different words and sentences.

Moreso than any other collection of tools discussed so far, there are a wide range of text mining and corpus analysis tools available. This is largely for two reasons: the first is that text mining is difficult, and English is a complicated language. Developing a complete suite of tools with broad application to different bodies of text and forms of media is an extremely difficult task. The diversity of tools for lexical analysis is a reflection of the diversity and complexity of the language that it seeks to measure and assess. The second reason is that different groups of linguistics researchers often have different approaches to describing and analyzing text, and the wide range of tools available for text mining is a result of multiple different fields of researchers constructing their own specific tools. We believe that the tools presented below represent a selection of tools that cut across the numerous facets of textual processing and analysis, and are suitable for general approaches to text mining as well as the investigation of specific constructs within text and discourse.

LIWC

The Linguistic Inquiry and Word Count (LIWC) tool (Tausczik & Pennebaker, 2010) is a graphical and easy-to-use computerized text analysis tool which measures the latent characteristics of a text through analysis of the vocabulary used. LIWC provides more than 80 metrics regarding different psychological categories of vocabulary (e.g., cognitive words, affective words, functional words, analytical words) and has been extensively used and validated in a large number of studies.

WMatrix

WMatrix (http://ucrel.lancs.ac.uk/wmatrix/) is an online graphical tool that can be used for word frequency analysis and visualization of text corpora. Although it can be used to conduct the complete analysis process, it is primarily useful in the feature engineering phase for extraction of linguistic features, including word n-grams, multi-word phrases such as idioms and similes (i.e., “take down a peg”), part-of-speech tags, and word semantic categories. It also provides visualization of the text corpora in the form of word clouds, and provides interface for comparison of several text corpora simultaneously.

Coh-Metrix

Another popular tool for text analysis is Coh-Metrix (Graesser, McNamara, & Kulikowich, 2011; Graesser, McNamara, Louwerse, & Cai, 2004) which provides more than 100 measures of text

divided into 11 categories. Compared to WMatrix, CohMetrix offers a more contextual understanding and analysis of text features and relationships in the data. Whereas WMatrix tags words and multi-word units semantically, CohMetrix has multiple tags for assessing deep text cohesion such as measures of narrativity, or referential cohesion. With these increases in the deep meanings of analysis comes a need for greater sized datasets – using CohMetrix effectively tends to require a larger corpus of text than semantic taggers.

Latent semantic analysis (LSA)

Another technique which is often used to extract topics from document corpora is Latent Semantic Analysis (Landauer, Foltz, & Laham, 1998). While LDA and similar probabilistic methods use word co-occurrence to estimate which words constitute a topic, LSA uses the linear algebra technique of matrix decomposition to find sets of words that represent different topics. It can be also used to measure the semantic similarity of two documents or parts of documents, by comparing their vectors in the topic space. LSA has been implemented in several programming languages, with a java-based text mining library (tml-java.sourceforge.net) and the lsa R package (Wild, 2015) being some of the most popular LSA implementations.

NLP toolkits (Stanford CoreNLP, Python NLTK, Apache OpenNLP)

Given that text mining systems typically involve analysis of natural language text, natural language processing (NLP) toolkits represent an important part of the text mining toolset. Those tools are typically used in the pre-processing stage of the analysis, for example, to a) split paragraphs into individual sentences, utterances, or words, b) extract syntactic dependencies between words, c) assign part-of-speech (word grammatical categories) categories to each word, d) reducing derived words to their root word (i.e., stemming and lemmatization), e) named-entity extraction, which is a process of finding named entities in the text (i.e., names of people, places, institutions, monetary amounts, dates), and f) co-reference resolution (resolution of pronouns to their target nouns). There are several NLP toolkits available, which provide programmable APIs for popular programming languages (e.g., Java and Python). One popular example is the Apache OpenNLP toolkit (Morton, Kottmann, Baldridge, & Bierner, 2005), a java-based NLP toolkit that supports most of the common NLP tasks listed above. Similarly, Python NLTK (Bird, 2006) is an NLP library for python programming language with very similar capabilities. Finally, Stanford CoreNLP (Manning et al., 2014) is an NLP toolkit which provides a Java API, a standalone command line interface, and a set of “wrappers” for other programming languages (e.g., C#, Python, R, Ruby, Scala, JavaScript).

ConceptNet

One of the primary reasons why understanding natural language is a very challenging problem is that each statement is heavily dependent on the particular context and background knowledge of the listener/reader. The approach taken by ConceptNet (Liu & Singh, 2004) is to develop an enormously large graph of “common sense” knowledge (e.g., “piano is a musical instrument”) which can be then utilized for understanding and processing natural text. By utilizing an extensive knowledge base, ConceptNet can be used to categorize textual documents, extract topical information from corpora, sentiment analysis (i.e., detecting emotions in the text), and summarization of text, among other uses.

TAGME
TAGME is a text annotation tool, specifically designed for semantic annotation of short, unstructured or semi-structured text segments, such as the text obtained from search-engine snippets, tweets or news feeds (Ferragina and Scaiella, 2010). The text annotation process identifies a sequence of terms and annotates them with pertinent links to Wikipedia pages. That is, TAGME assigns a Wikipedia concept to each of the term sequences in the analyzed text where possible. An experimental evaluation of TAGME (Ferragina and Scaiella, 2010) showed better performance on short text segments and a comparable precision/recall results on longer text, compared to other solutions. The tool provides an API for on-the-fly text processing and integration with other applications.

Apache Stanbol
Apache Stanbol is an open source software tool for semantic text analysis (stanbol.apache.org/docs/trunk/scenarios.html). It is primarily designed to bring semantic technologies into existing content management systems, and for text mining and feature extraction. Similar to TAGME, it links keywords extracted from text to Wikipedia concepts. Apache Stanbol is easy to set up and run on a small set of instances. However, the tool also allows for incorporating a domain specific ontology in the annotation process. This is highly beneficial when working with locally defined concepts specific to a given educational context. Finally, Apache Stanbol supports text annotation in multiple languages. The tool has been integrated with several content management systems.

2.4.3 Social Network Analysis

Social network analysis seeks to understand the connections and relationships that form between individuals and/or communities, most commonly expressed as node and edge diagrams. SNA is commonly employed to analyze collaborative social networks such as those seen in social media, or in student interaction within MOOCs or online courses.

Gephi (https://gephi.org) is a popular and widely used interactive tool for the analysis and visualization of different types of social networks. Gephi is extensively used in learning analytics research, and it supports directed and undirected social networks specified in a wide range of input data formats. Often used as a tool for exploratory analysis, it provides a set of graphical tools for easy visualization of social networks, including the ability to color nodes and edges based on their attributes of the properties of their network position (e.g., clustering coefficient, degree centrality, betweenness centrality). The tool also offers a Java API for manipulation of social network graphs, calculation of mutliple measures (e.g., density, average path, and betweenness centrality), and execution algorithms commonly used in social network analysis (e.g., graph clustering and giant connected component extraction). It is licensed under the GPL license and available on Microsoft Windows, Linux, and Mac OSX platforms.

EgoNet (http://egonet.sf.net) is a free social network analysis tool which focuses on the analysis of egocentric networks, which are, generally speaking, social networks constructed from the perspective of the individual network actors, typically using survey instruments. Through

EgoNet, a researcher specifies a set of network members and distributes to all of them a small survey regarding their relationships with other members of the network. As members provide information about network structure from their perspective (hence “ego” in the name), EgoNet visualizes the overall network structure and provides a set of analysis tools to better understand the overall network structure, with options to interrogate a member of the network with further questions.

NodeXL (Network Overview Discovery Exploration for Excel, http://nodexl.codeplex.com) is an extension for Microsoft Excel that makes it easy to visualize network data in Microsoft Excel from a wide variety of input data formats. Similarly to Gephi, it provides a set of tools for filtering and visualizing the data, and also the calculation of the basic network properties (e.g., radius, diameter, density), node properties (e.g., degree centrality, betweenness centrality, eigenvector centrality), and other network analysis methods (e.g., cluster analysis for community mining). Currently, there are two versions, NodeXL basic, which is free, and NodeXL Pro. Beyond basic support for social network analysis NodeXL Pro contains functionality for automated loading of data from several social media platforms (e.g., Twitter, YouTube, Flickr), and text and sentiment analysis of social media streams.

Pajek (http://mrvar.fdv.uni-lj.si/pajek) is a free desktop tool for complex analysis of a wide variety of large networks (thousands and hundreds of thousands of nodes), including the analysis of networks of social interactions. Pajek is extensively used in academia for social network analysis, including LA research, for tasks such as network partitioning, community detection, large network visualization, and information flow analysis. At present, Pajek is available for only Windows OS. There is also Pajek-XXL version which is a specially designed version of Pajek for working efficiently with extremely large networks (with millions of nodes or more).

NetMiner (http://www.netminer.com) is a commercial graphical tool for the analysis of networks and their visualizations. Similarly to Gephi and NodeXL, it supports importing network data in various formats, network visualizations, and calculation of common graph-based and node-based statistics. NetMiner is also suitable for advanced analyses of networks, and has a built-in data mining module supporting various data mining tasks (e.g., classification, clustering, recommendation, reduction). It also has an integrated Python scripting engine for more complex and custom types of analyses. Besides the graphical user interface, it also supports a scripting interface which makes it suitable for embedding as a module in other software systems. Finally, it supports 3D visualizations of networks and video recording of network explorations (e.g., for inclusions in presentations). NetMiner is currently available only on Microsoft Windows OS.

Cytoscape (http://www.cytoscape.org) is another open source platform, originally developed for the visualization of molecule interaction networks, which has become a fully-featured suite for analysis of various types of networks, including social networks. Cytoscape consists of a core distribution with basic network analysis and visualization capabilities, which is then extended using a large number of user-contributed modules. Cytoscape is developed on the Java platform and can be used within multiple operating systems.

SoNIA (https://web.stanford.edu/group/sonia) is an open source platform for analysis of longitudinal network data. In the case of longitudinal network data, besides information about relationships (i.e., edges) between network members (i.e., nodes), there is also information available about the time those relationships occurred or at least the order in which those relationships developed. SoNIA supports visualization of network changes over time, with the ability to specify different network layout algorithms to multiple timeframes to better visualize changes in network structure. The result is a nice “smooth” animation of structural changes over time, which can be exported into QuickTime video format. SoNIA is developed by Stanford University using the Java programming language and thus can be used in all major operating systems.

SocNetV (Social Networks Visualizer, http://socnetv.sourceforge.net) is an open-source tool for the analysis and visualization of social networks. It supports loading data from various network formats, calculation of typical graph and node properties, and flexible visualization of networked data (e.g., filtering, coloring, and resizing of nodes based on their properties). One interesting and unique feature of SocNetV is the embedded web crawler, which can be used to automatically extract a link structure between a collection of HTML documents. It is licensed under GPL license and available on Microsoft Windows, Linux, and Mac OSX platforms.

NetworkX (http://networkx.github.io) is an open source software library for the Python programming language for creation, manipulation, and analysis of complex network processes, structures and dynamics. It is heavily used in academia and provides a rich set of advanced functionalities for working with networked data, including graph reduction using block modeling techniques, graph clustering, community detection, link prediction (finding missing links, e.g., missing Facebook connection among two friends), network triads analysis, and others.

R packages: statnet (network, sna, ergm) and igraph. Aside from graphical tools for analysis of social networks, there are several packages for social network analysis in the R programming language. The **network package** is used for constructing and modifying network objects, extraction of simple network metrics, and visualization of network graphs. Often used together with the network package is the **sna package**, which contains a set of functionalities commonly needed for social network analysis, including calculation of network and node metrics, graph reduction using block modeling techniques, structural equivalence detection, network regression, graph generation, networks visualization, and others. Another package which is often used for social network analysis is the **igraph package**. It is a library written in the C programming language with additional language bindings for the R and Python programming languages. It can be used to construct and modify social networks from a wide variety of input formats (e.g., Pajek, Gephi, GraphML, edge list, and adjacency matrix), calculation of network and node properties, graph visualization, and for different network analyses including community detection, graph clustering, block modeling, calculation of cohesive blocks and others. Another important package for social network analysis is the **statnet** package, which focuses on statistical modeling of networks using exponential random graph models (ERGMs), latent space, and latent cluster models. The statnet package includes tools for network model

estimation, the evaluation of network models, model-based network simulations, and network visualization. It also includes and utilizes many of the other packages listed in this section, such as network, sna, and ergm.

SNAPP (The Social Networks Adapting Pedagogical Practice, https://github.com/aneesha/SNAPPVis) is a bookmarklet (i.e., a javascript program intended to be used as a button on the browser's bookmark bar) developed by Bakharia & Dawson (2011) for analysis of student social networks developed in common learning management systems - LMSs (e.g., Blackboard, Desire2Learn, and Moodle). SNAPP extracts a student social network (formed through students' posting and replying interactions) from HTML pages of LMS discussions. The data can be then exported for further analysis or visualized within SNAPP using several different graph layout algorithms or further analysis can be performed with other SNA tools discussed above. SNAPP can be also used to explore the evolution of student social networks across time, analysis of highly active/inactive users, identification of structural holes, and comparative analysis of several discussion forums.

2.4.4 Process and sequence mining

Besides more traditional approaches to educational data analysis, such as predicting learning outcome or course persistence, researchers also aim at tracking sequences of learner activities to understand learning strategies and processes (Bogarín, et al., 2014, Beheshitha et al., 2015). A distinctive set of tools has emerged for this type of application. In this section, we will introduce ProM and TraMineR - tools for process and sequence mining commonly used to support EDM and LA research. These tools are typically used for conducting analyses, though they also allow for some level of data pre-processing.

ProM (www.promtools.org/doku.php) is a Java based, platform independent, modular and open source platform that supports a wide variety of process mining techniques (Verbeek et al., 2010). The most recent implementation, ProM 6, supports running process mining in a distributed settings or through batch processing. ProM also supports chaining of several process mining algorithms, providing a clear specification of expected inputs and outputs for each of the supported implementations. Moreover, new plugins can be added at run-time, allowing for straightforward integration into the analysis process. Finally, ProM allows for easy integration with existing information systems without the need for programming.

TraMineR

TraMineR (http://traminer.unige.ch) is a free and open source R-package that supports mining and visualizing state or event sequences. Some of the primary features of TraMineR for the analysis and visualization of state sequence data include: i) processing different formats of state sequences and transforming to and from various representations, ii) describing longitudinal (e.g., length, complexity, time in each state) and other aggregated characteristics of sequences, iii) access to a wide variety of plotting capabilities (e.g., frequency or density plots, index plot), and iv) a broad set of metric for evaluating distances between sequences.

2.5 PSLC DataShop

A final tool examined in this review paper is the multi-functional PSLC DataShop (https://pslcdatashop.web.cmu.edu/, Koedinger et al., 2010). The PSLC DataShop consists of a repository of many data sets that is available to download and analyze, as well as a collection of tools to support exploratory analyses and models. DataShop has functionality for comparing domain structure (knowledge component) models, including q-matrices (Tatsuoka, 1983), on a data set. It also has the ability to visualize student performance over time in terms of correctness, hint use, latent knowledge, response times, and other variables of interest. Additionally, it offers visualizations of student performance, at an item-by-item level. The PSLC DataShop is a web application, available for free, but not open-source.

3 Summary

In this article, we have reviewed 40 tools frequently used for data mining/analytics in the area of education. This is a rapidly changing area, and new tools are emerging constantly. Nonetheless, we hope that this review will prove useful to researchers interested in learning about these emerging methods not just at a theoretical level, but in terms of practical application and use.

One key consideration for researchers and practitioners new to educational data mining and learning analytics is that no one tool is ideally suited to conducting the entire process of analyzing most data sets from start to finish. Different tools are uniquely suited to different tasks. For example, a researcher may have data on 60 million system transactions in a popular MOOC. From this dataset he or she wishes to select only data of a particular year (SQL), then refine that dataset to calculate total student time in the system (Excel) before fitting a predictive model (RapidMiner) that analyzes the relationship between forum posts and replies (NodeXL) and overall textual quality of posts and replies by that student (CohMetrix). Finally, this researcher may wish to visualize the most interesting clusters of students found within the social network data (Gephi).

These tools form part of a collection – a toolbox – that researchers in the fields of EDM and LA currently use. No researcher (that we are aware of) uses all of these tools, but they are represented in aggregate across the different groups of scientists working in this field. They represent different approaches to different problems, each with their own particular strengths and weaknesses. Through using a combination of tools, complex analyses are realized, and useful discoveries can be made.

References

Baker, R.S. (2015) *Big Data and Education*. 2nd Edition. New York, NY: Teachers College, Columbia University.

Baker, R.S.J.d. (2007) Modeling and Understanding Students' Off-Task Behavior in Intelligent Tutoring Systems. *Proceedings of ACM CHI 2007: Computer-Human Interaction*, 1059-1068.

Baker, R.S.J.d., Corbett, A.T., Wagner, A.Z. (2006) Human Classification of Low-Fidelity Replays of Student Actions. *Proceedings of the Educational Data Mining Workshop at the 8th International Conference on Intelligent Tutoring Systems*, 29-36.

Baker, R., Siemens, G. (2014) Educational data mining and learning analytics. In Sawyer, K. (Ed.) *Cambridge Handbook of the Learning Sciences: 2nd Edition*, pp. 253-274.

Bakharia, A., & Dawson, S. (2011, February). SNAPP: a bird's-eye view of temporal participant interaction. In *Proceedings of the 1st international conference on learning analytics and knowledge* (pp. 168-173). ACM.

Berry, M. J., & Linoff, G. (1997). *Data mining techniques: for marketing, sales, and customer support*. John Wiley & Sons, Inc.

Beheshitha, S. S., Gašević, D., & Hatala, M. (2015, March). A process mining approach to linking the study of aptitude and event facets of self-regulated learning. In *Proceedings of the Fifth International Conference on Learning Analytics and Knowledge* (pp. 265-269). ACM.

Bird, S. (2006). NLTK: the natural language toolkit. In Proceedings of the COLING/ACL on Interactive presentation sessions (pp. 69–72). Association for Computational Linguistics.

Blei, D. M. (2012). Probabilistic Topic Models. *Commun. ACM*, *55*(4), 77–84. http://doi.org/10.1145/2133806.2133826

Bogarín, A., Romero, C., Cerezo, R., & Sánchez-Santillán, M. (2014, March). Clustering for improving educational process mining. In *Proceedings of the Fourth International Conference on Learning Analytics And Knowledge* (pp. 11-15). ACM.

Cetintas, S., Si, L., Xin, Y. P., & Hord, C. (2010). Automatic detection of off-task behaviors in intelligent tutoring systems with machine learning techniques. *Learning Technologies, IEEE Transactions on*, *3*(3), 228-236.

Chang, J. (2010). Package “lda.” Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.216.2273&rep=rep1&type=pdf

Collins M, Schapire RE, Singer Y (2004) Logistic regression, adaboost and Bregman distances. Mach Learn
48:253–285

Corbett, A. T., & Anderson, J. R. (1995). Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, *4*(4), 253-278.

M. Cornolti, P. Ferragina, and M. Ciaramita, “A Framework for Benchmarking Entity-annotation Systems,” in Proceedings of the 22Nd International Conference on World Wide Web, Republic and Canton of Geneva, Switzerland, 2013, pp. 249–260.

Duval, E. (2011, February). Attention please!: learning analytics for visualization and recommendation. In *Proceedings of the 1st International Conference on Learning Analytics and Knowledge* (pp. 9-17). ACM.

Ferragina, P., & Scaiella, U. (2010, October). TAGME: onthe-fly annotation of short text fragments (by wikipedia entities). In Proceedings of the 19th ACM international conference on Information and knowledge management (pp. 1625-1628). ACM.

Graesser, A. C., McNamara, D. S., & Kulikowich, J. M. (2011). Coh-metrix providing multilevel analyses of text characteristics. *Educational Researcher*, *40*(5), 223-234.

Graesser, A. C., McNamara, D. S., Louwerse, M. M., & Cai, Z. (2004). Coh-Metrix: Analysis of text on cohesion and language. *Behavior research methods, instruments, & computers*, *36*(2), 193-202.

Grun, B., & Hornik, K. (2014). topicmodels: An R Package for Fitting Topic Models.

Jean-Louis, L., Zouaq, A., Gagnon, M., & Ensan, F. (2014). An assessment of online semantic annotators for the keyword extraction task. In *PRICAI 2014: Trends in Artificial Intelligence* (pp. 548-560). Springer International Publishing.

Jovanovic, J., Bagheri, E., Cuzzola, J., Gasevic, D., Jeremic, Z., & Bashash, R. (2014). Automated Semantic Tagging of Textual Content. *IT Professional*, *16*(6), 38-46.

Koedinger, K. R., Baker, R. S., Cunningham, K., Skogsholm, A., Leber, B., & Stamper, J. (2010). A data repository for the EDM community: The PSLC DataShop. *Handbook of educational data mining*, *43*.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An introduction to latent semantic analysis. Discourse Processes, 25(2-3), 259–284. http://doi.org/10.1080/01638539809545028

Liu, H., & Singh, P. (2004). ConceptNet—a practical commonsense reasoning tool-kit. BT Technology Journal, 22(4), 211–226.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations (pp. 55–60).

McCallum, A. K. (2002). *MALLET: A Machine Learning for Language Toolkit*.

Morton, T., Kottmann, J., Baldridge, J., & Bierner, G. (2005). Opennlp: A java-based nlp toolkit.

Preacher, K.J., Curran, C.J., Bauer, D.J. (2006) *Journal of Educational and Behavioral Statistics, 31* (3), 437-448.

Reich, J., Tingley, D., Leder-Luis, J., Roberts, M. E., & Stewart, B. (2014). Computer-Assisted Reading and Discovery for Student Generated Text in Massive Open Online Courses. *Journal of Learning Analytics*, *2*(1), 156–184.

Reye, J. (2004). Student modelling based on belief networks. *International Journal of Artificial Intelligence in Education*, *14*(1), 63-96.

Rodrigo, M., Mercedes, T., d Baker, R. S., McLaren, B. M., Jayme, A., & Dy, T. T. (2012). Development of a Workbench to Address the Educational Data Mining Bottleneck. *International Educational Data Mining Society*.

Romero, C., & Ventura, S. (2010). Educational data mining: a review of the state of the art. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, *40*(6), 601-618.

Romero, C., & Ventura, S. (2007). Educational data mining: A survey from 1995 to 2005. *Expert systems with applications*, *33*(1), 135-146.

Tatsuoka, K. K. (1983). Rule space: An approach for dealing with misconceptions based on item response theory. *Journal of Educational Measurement*, *20*(4), 345-354.

Tausczik, Y. R., & Pennebaker, J. W. (2010). The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of language and social psychology*, *29*(1), 24-54.

Tervakari, A. M., Silius, K., Koro, J., Paukkeri, J., & Pirttila, O. (2014, April). Usefulness of information visualizations based on educational data. In *Global Engineering Education Conference (EDUCON), 2014 IEEE* (pp. 142-151). IEEE.

Tukey, J. W. (1977). Exploratory data analysis.

Veeramachaneni, K., Adl, K., & O'Reilly, U. M. (2015). Feature factory: Crowd sourced feature discovery. In *Proceedings of the Second (2015) ACM Conference on Learning@ Scale* (pp. 373-376). ACM.

Verbeek, H. M. W., Buijs, J. C. A. M., Van Dongen, B. F., & van der Aalst, W. M. (2010). Prom 6: The process mining toolkit. Proc. of BPM Demonstration Track, 615, 34-39.

Verbert, K., Duval, E., Klerkx, J., Govaerts, S., & Santos, J. L. (2013). Learning analytics dashboard applications. *American Behavioral Scientist*, 0002764213479363.

Wang, J. T., Zaki, M. J., Toivonen, H. T., & Shasha, D. (2005). *Introduction to Data Mining in Bioinformatics* (pp. 3-8). Springer London.

Wild, F. (2015). lsa: Latent Semantic Analysis. Retrieved from https://CRAN.R-project.org/package=lsa

Witten, I., H., Frank, E., Hall, M., A. (2011). Data mining: practical machine learning tools and techniques.